LEARNING WITH FEATURE DEPENDENT LABEL NOISE: A PROGRESSIVE APPROACH

Anonymous authors
Paper under double-blind review

ABSTRACT

Label noise is frequently observed in real world large scale datasets. The noise is introduced due to a variety of reasons; it is heterogeneous and feature-dependent. Most existing approaches to handle noisy labels fall into two categories: they either assume an ideal feature-independent noise, or remain heuristic without theoretical guarantees. In this paper, we propose to target a new family of feature-dependent label noise, which is much more general than commonly used i.i.d. label noise and encompasses a broad spectrum of noise patterns. Focusing on this general noise family, we propose a progressive label correction algorithm that iteratively corrects labels and refine the model. We provide theoretical guarantees showing that for a wide variety of (unknown) noise patterns, a classifier trained with this strategy converges to be consistent with the Bayes classifier. In experiments, our method outperforms SOTA baselines and is robust to various noise types and levels.

1 INTRODUCTION

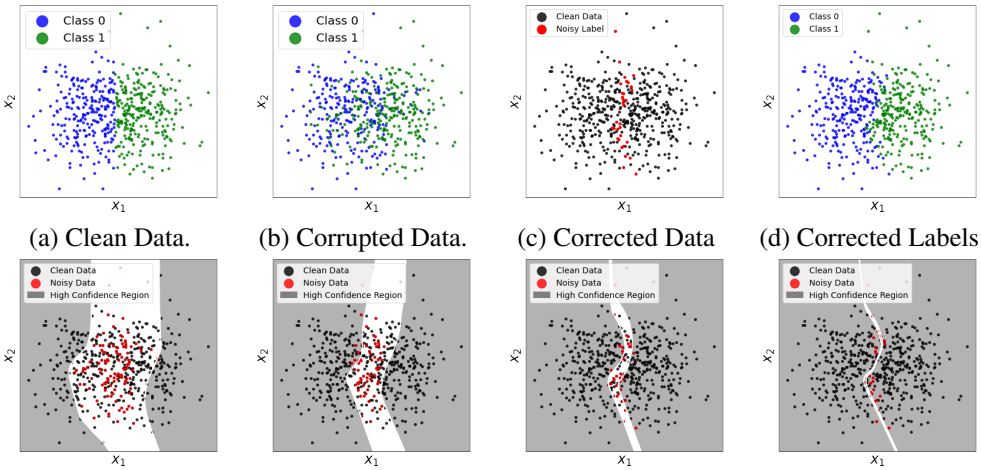
Addressing noise in training set labels is an important problem in supervised learning. Incorrect annotation of data is inevitable in large-scale data collection, due to intrinsic ambiguity of data/class and mistakes of human/automatic annotators (Yan et al., 2014; Veit et al., 2017). Developing methods that are resilient to label noise are therefore crucial in real life applications.

Classical approaches take a rather simplistic *i.i.d. assumption* of the label noise, i.e., the label corruption is independent and identically distributed and thus is feature independent. Methods based on this assumption either explicitly estimate the noise pattern (Reed et al., 2014; Patrini et al., 2017; Dan et al., 2019; Xu et al., 2019) or introduce extra regularizer/loss terms (Natarajan et al., 2013; Van Rooyen et al., 2015; Xiao et al., 2015; Zhang & Sabuncu, 2018; Shen & Sanghavi, 2019). Some results prove that the commonly used losses are naturally robust against such i.i.d. label noise (Manwani & Sastry, 2013; Ghosh et al., 2015; Gao et al., 2016; Ghosh et al., 2017; Charoenphakdee et al., 2019; Hu et al., 2020).

Although these methods come with theoretical guarantees, they usually do not perform as well as expected in practice due to the unrealistic i.i.d. assumption on noise. This is likely because *label noise is heterogeneous and feature dependent.* A cat with an intrinsically ambiguous appearance is more likely to be mislabeled as a dog. An image with poor lighting or severe occlusion can be mislabeled, as important visual clues are imperceptible. Methods that can combat label noise of a much more general form are very much needed to address real world challenges.

To be more adaptive to the heterogeneous label noise, state-of-the-art (SOTA) methods often resort to a *data-recalibrating* strategy. They progressively identify trustworthy data or correct data labels, and then train using these data (Tanaka et al., 2018; Wang et al., 2018; Jiang et al., 2018). The models gradually improve as more clean data are collected or more labels are corrected, eventually converging to models of high accuracy.

These data-recalibrating methods best leverage the learning power of deep neural nets and achieve superior performance in practice. However, their underlying mechanism remains a mystery. No methods in this category can provide theoretical insights as to why the model can converge to an ideal one. As a consequence, these methods require careful tuning on hyperparameters such as the pace and criterion of data collection, and thus are very hard to generalize.



(a) Clean Data. (b) Corrupted Data. (c) Corrected Data (d) Corrected Labels

(e) Epoch 10 (f) Epoch 20 (g) Epoch 30 (h) Final Epoch

Figure 1: Illustration of the algorithm using synthetic data. (a) Gaussian blob with clean label ($\eta^*(\boldsymbol{x})$). (b) Data with corrupted labels. (c) Final corrected data. Black dots are all data that have their clean label. Red dots are the data that remain incorrect. that remain un-corrected and are closer to the decision boundary. Our algorithm corrects most of the noise only using noisy classifier's confidence. (d) Data set after label correction. (e)-(h) We show the intermediate results at different iterations. Gray region is the area where the classifier has high confidence. Labels within this region are corrected.

In this paper, we propose a novel and principled method that specifically targets the heterogeneous, feature-dependent label noise. Unlike previous methods, we target a much more general family of noise, called *Polynomial Margin Diminishing* (PMD) label noise. In this noise family, we allow arbitrary noise level except for data far away from the true decision boundary. This is consistent with the real-world scenario; data near the decision boundary are harder to distinguish and more likely to be mislabeled. Meanwhile, a datum far away from the decision boundary is a typical example of its true class and should have a reasonably bounded noise level.

Assuming this new PMD noise family, we propose a theoretically-guaranteed data-recalibrating algorithm that gradually corrects labels based on the noisy classifier's confidence. We start from data points with high confidence, and correct the labels of these data using the predictions of the noisy classifier. Next, the model is improved using the cleansed label. We continue alternating the label correction and model improvement until it converges. See Figure 1 for an illustration. Our main theorem shows that with theory-informed choice of criterion for label correction at each iteration, the improvement of the label purity is guaranteed. Thus the model is guaranteed to improve with sufficient rate through iterations and eventually becomes consistent with the Bayes optimal classifier.

Beside the theoretical strength, we also demonstrate the power of our method in practice. Our method outperforms others on CIFAR-10/100 with various synthetic noise patterns. We also evaluate our method against SOTA on a real-world dataset with unknown noise pattern.

To the best of our knowledge, our method is the first data-recalibrating method that is theoretically guaranteed to converge to an ideal model. The PMD noise family encompasses a broad spectrum of heterogeneous, feature dependent noise and better approximates the real-world scenario. It also provides a novel theoretical setting for the study of label noise.

Related works. We review works that do not assume an i.i.d. label noise. Menon et al. (2018) generalized the work of (Ghosh et al., 2015) showing that certain loss (e.g., zero-one and ramp loss) are naturally robust to specific types of feature-dependent label noise. However, their method does not recalibrate individual data based on their contexts, and thus are not as effective as other deep-learning-based methods in practice. Cheng et al. (2020) proposed an active learning method to iteratively query the clean labels from an oracle. This approach is not applicable to settings when kosher annotations are inaccessible.

Data-recalibrating methods use noisy networks' predictions to iteratively select/correct data and improve the models. Tanaka et al. (2018) introduced a loss to perform label correction based on the

network historical predictions. Wang et al. (2018) identified noisy labels as outliers based on their label consistency with surrounding data. Jiang et al. (2018) used a separate network trained on a small kosher data set to determine if a datum is clean. (Yu et al., 2019; Han et al., 2018) trained two synchronized networks. The confidence and consistency of the two networks are used to select clean data. For completeness, we also refer to other methods of similar design (Li et al., 2017; Vahdat, 2017; Veit et al., 2017; Ma et al., 2018; Thulasidasan et al., 2019; Arazo et al., 2019; Shu et al., 2019; Yi & Wu, 2019). As for theoretical guaratees, Ren et al. (2018) proved the convergence of the training, but cannot guarantee that the model converges to an ideal one. Zheng et al. (2020) proved an one-shot guarantee for their data-recalibrating method. But the convergence of the model is not guaranteed. *Our method is the first data-recalibrating method which is guaranteed to converge to a well behaved classifier.*

2 METHOD

We start by introducing the family of Poly-Margin Diminishing (PMD) label noise. In Section 2.2, we present our main algorithm. Finally, we prove the correctness of our algorithm in Section 3.

Notations and preliminaries. Although the noise setting and algorithm naturally generalize to multi-class, for simplicity we focus on binary classification. Let the feature space be $\mathcal{X}$. We assume the data $(\boldsymbol{x}, y)$ is sampled from an underlying distribution D on $\mathcal{X} \times \{0, 1\}$. Define the posterior probability $\eta(\boldsymbol{x}) = \mathbb{P}[y = 1 \mid \boldsymbol{x}]$. Let $\tau_{0,1}(\boldsymbol{x}) = \mathbb{P}[\widetilde{y} = 1 \mid y = 0, \boldsymbol{x}]$ and $\tau_{1,0}(\boldsymbol{x}) = \mathbb{P}[\widetilde{y} = 0 \mid y = 1, \boldsymbol{x}]$ be the noise functions, where $\widetilde{y}$ denotes the corrupted label. For example, if a datum $\boldsymbol{x}$ has true label $y = 0$, it has $\tau_{0,1}(\boldsymbol{x})$ chance to be corrupted to 1. Similarly, it has $\tau_{1,0}(\boldsymbol{x})$ chance to be corrupted from 1 to 0. Let $\widetilde{\eta}(\boldsymbol{x}) = \mathbb{P}[\widetilde{y} = 1 \mid \boldsymbol{x}]$ be the noisy posterior probability of $\widetilde{y} = 1$ given feature $\boldsymbol{x}$. Let $\eta^*(\boldsymbol{x}) = \mathbb{I}_{\{\eta(\boldsymbol{x}) \geq \frac{1}{2}\}}$ be the (clean) Bayes optimal classifier, where $\mathbb{I}_A$ equals 1 if A is true, and 0 otherwise. Finally, let $f(\boldsymbol{x}) : \mathcal{X} \to [0, 1]$ be the classifier scoring function (the softmax output of a neural network in this paper).

2.1 POLY-MARGIN DIMINISHING NOISE

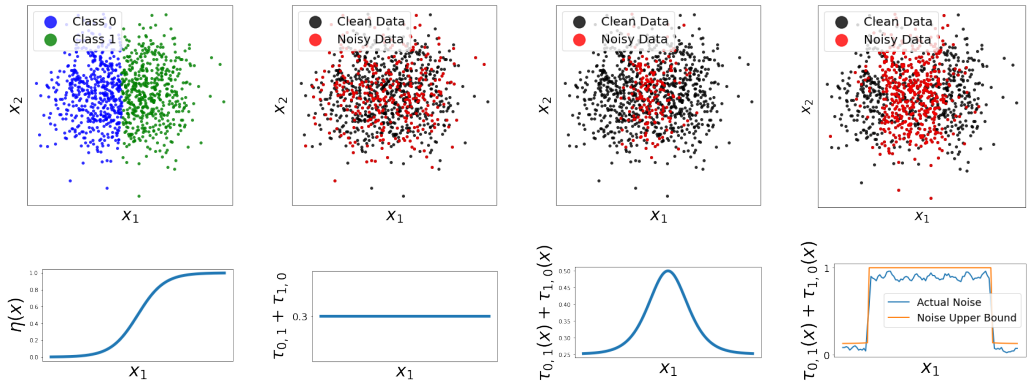
We first introduce the family of noise functions τ this paper will address. We introduce the concept of *polynomial margin diminishing noise* (PMD noise), which only upper bounds the noise τ in a certain level set of $\eta(\boldsymbol{x})$, thus allowing τ to be arbitrarily high outside the restricted domain. This formulation not only covers the feature-independent scenario but also generalizes scenarios proposed by (Du & Cai, 2015; Menon et al., 2018; Cheng et al., 2020).

Definition 1 (PMD noise)**.** *A pair of noise functions $\tau_{0,1}(\boldsymbol{x})$ and $\tau_{1,0}(\boldsymbol{x})$ are polynomial-margin diminishing (PMD), if there exist constants $t_0 \in (0, \frac{1}{2})$, and $c_1, c_2 > 0$ such that:*

$$\begin{aligned} \tau_{1,0}(\boldsymbol{x}) \leq c_1[1 - \eta(\boldsymbol{x})]^{1+c_2}; \forall \eta(\boldsymbol{x}) \geq \frac{1}{2} + t_0, \text{ and} \\ \tau_{0,1}(\boldsymbol{x}) \leq c_1 \eta(\boldsymbol{x})^{1+c_2}; \forall \eta(\boldsymbol{x}) \leq \frac{1}{2} - t_0. \end{aligned} \tag{1}$$

We abuse notation by referring to t_0 as the "margin" of τ. Note that the PMD condition only requires the *upper bound* on τ to be polynomial and monotonically decreasing in the region where the Bayes classifier is fairly confident. For the region $\{\boldsymbol{x} : |\eta(\boldsymbol{x}) - \frac{1}{2}| < t_0\}$, we allow both $\tau_{01}(x)$ and $\tau_{10}(x)$ to be arbitrary. Figure 2(d) illustrate the upperbound (red curve) and a sample noise function (blue curve). We also show the corrupted data according to this noise function (red points are the corrupted labels whereas black points are clean labels).

The PMD noise family is much more general than existing noise assumptions. For example, the boundary consistent noise (BCN) (Du & Cai, 2015; Menon et al., 2018) assumes a noise function that monotonically decreases as the data is moving away from the decision boundary. See Figure 2(c) for an illustration. This noise is much more restrictive compared to our PMD noise which (1) only requires a monotonic upper bound, and (2) allows arbitrary noise strength in a wide buffer near the decision boundary. Figure 2(b) shows a traditional feature independent noise pattern (Reed et al., 2014; Patrini et al., 2017), which assumes $\tau_{0,1}(\boldsymbol{x})$ (resp. $\tau_{1,0}(\boldsymbol{x})$) to be a constant independent of $\boldsymbol{x}$.



(a) Clean labels (b) Uniform noise (c) BCN noise (d) Diminishing noise

Figure 2: An illustration of different noise functions. (a) The original data: Gaussian blobs with clean labels (by clean label, we refer to the prediction of the Bayes classifier $\eta^*(\boldsymbol{x})$, not y). Confident region of η (and thus f) in this case is place where $\eta(\boldsymbol{x})$ is closed to 0 or 1. Blue and green dots are two classes. (b) Uniform label noise: each point has a equal probability to be flipped. Red dots are data with corrupted labels. Black dots are data that was not corrupted. (c) BCN noise: the level of noise is decreasing as $\eta^*(\boldsymbol{x})$ becomes confident (d) PMD noise: noise level (blue) is only upper bounded by diminishing polynomial function when $\eta(\boldsymbol{x})$ is higher or lower than certain threshold. The upper bound is drawn in red curve.

2.2 The Progressive Correction Algorithm

Our algorithm iteratively trains a neural network and corrects labels. We start with a warm-up period, in which we train the neural network (NN) with the original noisy data. This allows us to attain a reasonable network before it starts fitting noise (Zhang et al., 2017). After the warm-up period, the classifier can be used for label correction. We only correct a label on which the classifier f has a very high confidence. The intuition is that under the noise assumption, there exists a "pure region" in which the prediction of the noisy classifier f is highly confident and is consistent with the clean Bayes optimal classifier η. Thus the label correction gives clean labels within this pure region. In particular, we select a high threshold θ. If f predicts a different label as $\tilde{y}$ and its confidence is above the threshold, $|f(\boldsymbol{x}) - 1/2| > \theta$, we flip the label $\tilde{y}$ to the prediction of f. We repeatedly correct labels and improve the network until no label is corrected. Next, we slightly decrease the threshold θ, use the decreased threshold for label correction, and improve the model accordingly. We continue the process until convergence. For convenience in theoretical analysis, in the algorithm, we define a continuous increasing threshold T and let $\theta = 1/2 - T$. Our algorithm is summarized in Algorithm 1. In Section 3, we will show that this iterative algorithm will converge to be consistent with clean Bayes classifier $\eta^*(\boldsymbol{x})$ for most of the input instances.

Generalizing to the multi-class algorithm. In multi-class scenario, denote by $f_i(\boldsymbol{x})$ the classifier's prediction probability of label i. Let $h_{\boldsymbol{x}}$ be the classifier's class prediction, i.e., $h_{\boldsymbol{x}} = \operatorname{argmax}_i f_i(\boldsymbol{x})$. We change the $\left|f(\boldsymbol{x}) - \frac{1}{2}\right|$ term to the gap between the highest confidence $f_{h_{\boldsymbol{x}}}(\boldsymbol{x})$ and the confidence on $\widetilde{y}$, $f_{\widetilde{y}}(\boldsymbol{x})$. If the absolute different between these two confidence is bigger than certain threshold θ, then we correct $\widetilde{y}$ to $h_{\boldsymbol{x}}$. In practice, we find using the difference of logarithms will be more robust.

3 Analysis

Our analysis focuses on the asymptotic case and answers the following question: Given infinitely many noisy labeled data, is it possible to learn a reasonably good classifier? We show that if the noise satisfies the arguably-general PMD condition, the answer is yes. Assuming mild conditions on the hypothesis class of the machine learning model and the distribution D, we prove that Algorithm 1 obtains a nearly clean classifier. This reduces the challenge of noisy label learning from a realizable problem into a sample complexity problem. In this work we only focus on the asymptotic case, and leave the sample complexity for future work.

Algorithm 1 `Progressive Label Correction`

Input: Dataset: $\tilde{S} = \{(\boldsymbol{x}_1, \widetilde{y}_1^0), ..., (\boldsymbol{x}_n, \widetilde{y}_n^0)\}$, Initial NN:$f(\boldsymbol{x})$
Step size:β, Initial and End Threshold (T_0, T_{end}), Warm-up: m, Total round N;
Output: $f_{final}(\cdot)$

```
T = T_0
for t = 1, ··· , N do
    Train f(x) on S̃
    if t ≥ m then
        θ = 1/2 − T
        for all (x, ỹ^{t−1}) ∈ S̃, |f(x) − 1/2| ≥ θ do
            ỹ^t = 𝟙_{{f(x) ≥ 1/2}}
        end for
        if ∀i ∈ [1, ··· , n], ỹ_i^t = ỹ_i^{t−1} then
            T = min(T(1 + β), T_end)
        end if
    end if
    S̃ = {(x_1, ỹ_1^t), ..., (x_n, ỹ_n^t)}
end for
```

3.1 Assumptions:

Our first assumption restricts the model to be able to at least approximate the true Bayes classifier $\eta(\boldsymbol{x})$. This condition assumes that given a hypothesis class $\mathcal{H}$ with sufficient complexity, the approximation gap between a classifier $f(x)$ in this class and $\eta(x)$ is determined by the inconsistency between the noisy labels and the optimal Bayes classifier.

Definition 2 (Level set (α, ϵ) consistency)**.** *Suppose data are sampled as $(x, \tilde{y}) \sim D(\boldsymbol{x}, \tilde{\eta}(\boldsymbol{x}))$ and $f(\boldsymbol{x}) = \arg\min_{h \in \mathcal{H}} E_{(x,\tilde{y})\sim D(\boldsymbol{x},\tilde{\eta}(\boldsymbol{x}))} Loss(h(\boldsymbol{x}), \tilde{y})$. Given $\varepsilon < \frac{1}{2}$, we call $\mathcal{H}$ is (α, ϵ)-consistent if:*

$$|f(\boldsymbol{x}) - \eta(\boldsymbol{x})| \leq \alpha \mathbb{E}_{(\boldsymbol{z},\tilde{y})\sim D(\boldsymbol{z},\tilde{\eta}(\boldsymbol{z}))} \left[\mathbb{1}_{\{\tilde{y}_z \neq \eta^*(\boldsymbol{z})\}}(\boldsymbol{z}) \left| \left| \eta(\boldsymbol{z}) - \frac{1}{2} \right| \geq \left| \eta(x) - \frac{1}{2} \right| \right| \right] + \epsilon \quad (2)$$

For two input instances $\boldsymbol{z}$ and $\boldsymbol{x}$ such that $\eta(\boldsymbol{z}) > \eta(\boldsymbol{x})$ (and hence the Bayes classifier $\eta(\boldsymbol{x})^*$ has higher confidence at $\boldsymbol{z}$ than it does at $\boldsymbol{x}$), the indicator function $\mathbb{1}_{\{\tilde{y}_z \neq \eta^*(\boldsymbol{z})\}} \left(\boldsymbol{z} : \left|\eta(\boldsymbol{z}) - \frac{1}{2}\right| \geq \left|\eta(x) - \frac{1}{2}\right|\right)$ equals to 1 if the more confident point $\boldsymbol{z}$'s label is inconsistent with $\eta(\boldsymbol{x})^*$. This condition says that the approximation error of the classifier at $\boldsymbol{x}$ should be controlled by the risk of $\eta^*(.)$ at points $\boldsymbol{z}$ where $\eta^*(.)$ is more confident than it is at $\boldsymbol{x}$.

We next define a regularity condition of data distribution which describes the continuity of the level set density function.

Definition 3 (Level set bounded distribution)**.** *Define the margin $t(x) = |\eta(\boldsymbol{x}) - \frac{1}{2}|$ and $G(t)$ be the cdf of t: $G(t) = \mathbb{P}_{x\sim D}(|\eta(\boldsymbol{x}) - \frac{1}{2}| \leq t)$. Let $g(t) = G'(t)$ be the density function of t. We say the distribution D is (c_*, c^*)-bounded if for all $0 \leq t \leq 1/2$, $0 < c_* < g(t) \leq c^*$. If D is (c_*, c^*)-bounded, we define the worst-case density-imbalance ratio of D by ℓ_D:= $\frac{c^*}{c_*}$.*

The above condition enforces the continuity of Level set density function. This is crucial in the analysis since such continuity allows one to borrow information from its neighborhood region so that a cleaned neighbor can help correct the corrupted label. To simplify notation, we will omit D in the subscript when we mention ℓ. From now on, we will assume:

Assumption 1: There exist constants $\alpha, \epsilon, c_*, c^*$ such that the hypothesis class $\mathcal{H}$ is (α, ϵ)-consistent and the unknown distribution D is (c_*, c^*)-bounded.

3.2 Main Result and Proof Sketch

In this section we first state our main result, and present the supporting claims. Complete proofs can be found in the appendix. Our main result below states that if our starting function is trained

correctly, i.e., $f(\boldsymbol{x}) = \arg\min_{h\in\mathcal{H}} E_{(x,\tilde{y})\sim D(\boldsymbol{x},\tilde{\eta}(\boldsymbol{x}))} Loss(h(\boldsymbol{x}), \tilde{y})$, then Algorithm 1 terminates with most of the final labels matching the optimal Bayes classifier labels. In practice, minimizing true risk is not achievable. Instead, the empirical risk is used to estimate true risk, which approaches true risk asymptotically. For a scoring function f, we will denote by $y_{f(\boldsymbol{x})} := \mathbb{I}(f(\boldsymbol{x}) \geq 1/2)$ the label predicted by f.

Theorem 1. *Under assumption 1, for any noise τ which is PMD with margin t_0, define $e_0 = max(t_0, \frac{\alpha+\varepsilon}{1+2\alpha})$. Then the output of Algorithm 1 with f as above and with the following initializations: a) $T_0 < \frac{1}{2} - e_0$, b) $m \geq \frac{\ell\alpha}{\varepsilon}\log(\frac{2T_0}{1-2e_0})$, c)$N \geq m + \frac{1}{\beta}\log(\frac{T_0}{3\varepsilon})$, d) $T_{end} \leq 3\epsilon$ and e) $\beta \leq \frac{\varepsilon}{\alpha\ell}$, we have*

$$\mathbb{P}_{x\sim D}[y_{f_{final}(\boldsymbol{x})} = \eta^*(\boldsymbol{x})] \geq 1 - 3c^*\epsilon.$$

In the remainder of this section we shall assume that the noise τ is PMD with margin t_0. To prove our result we first define a "pure" level set.

Definition 4 (Pure (e, f, η)-level set)**.** *A set $L(e, \eta) := \{x | |\eta(\boldsymbol{x}) - \frac{1}{2}| \geq e\}$ is pure for f if $y_{f(\boldsymbol{x})} = \eta^*(\boldsymbol{x})$ for all $\boldsymbol{x} \in L(\eta, e)$.*

We now state a lemma that forms the foundation of our progressive correction algorithm. We show that given a tiny region where the model is reliable, we can move one step forward by trusting the model. Although the improvement is slight in a single round, it empowers a conservatively recursive step in the Algorithm 1.

Lemma 1 (One round purity improvement)**.** *Assume 1), and an f such that there exists a pure (e, f, η)-level set with $3\epsilon \leq e < \frac{1}{2}$. Let $\tilde{\eta}_{new}(\boldsymbol{x}) = y_{f(\boldsymbol{x})}$ if $|f(\boldsymbol{x}) - 1/2| \geq e$ and $\tilde{\eta}(\boldsymbol{x})$ if $|f(\boldsymbol{x}) - 1/2| < e$, and assume $f_{new} = \arg\min_{h\in\mathcal{H}} E_{(x,\tilde{y})\sim D(\boldsymbol{x},\tilde{\eta}_{new}(\boldsymbol{x}))} Loss(h(\boldsymbol{x}), \tilde{y})$. Let $e_{new} = \min\{e | e > 0,\ L(e, \eta)$ is pure for $f_{new}\}$. Then $\frac{1}{2} - e_{new} \geq (1 + \frac{\varepsilon}{\alpha\ell})(\frac{1}{2} - e)$.*

The above lemma states that the cleaned region will be enlarged by at least a constant factor. In following lemma, we justify the functionality of first m warm-up rounds. Since the initial neural network can behave badly, the region where we can trust the classifier can be very limited. Before considering starting the flipping procedure in a relatively larger level set, one first needs to expand the initial tiny region $\frac{1}{2} - e_0$ to a constant T_0.

Lemma 2 (Warm-up Rounds)**.** *Suppose for a given function f_0 there exists a level set $L(e_0, \eta)$ which is pure for f_0. Given $T_0 < 1/2$, after running Algorithm 1 for $m \geq \frac{\ell\alpha}{\varepsilon}\log(\frac{2T_0}{1-2e_0})$ rounds, there exists a level set $L(\frac{1}{2} - T_0, \eta)$ that is pure for f_0.*

Next we present our final lemma that combines the previous two lemmata.

Lemma 3. *Assume 1), and suppose for a given function f_0 there exists a level set $L(e_0, \eta)$ which is pure for f_0. If one runs Algorithm 1 starting with f_0 and the initializations: a) $T_0 < \frac{1}{2} - e_0$, b) $m \geq \frac{\ell\alpha}{\varepsilon}\log(\frac{2T_0}{1-2e_0})$, c) $N \geq m + \frac{1}{\beta}\log(\frac{1-6\varepsilon}{2T_0})$, d) $T_{end} \leq \frac{1}{2} - 3\epsilon$ and e) $\frac{\varepsilon}{\alpha\ell} \leq \beta \leq \frac{2\varepsilon}{\alpha\ell}$. Then $\mathbb{P}_{x\sim D}[y_{f_{final}(\boldsymbol{x})} = \eta^*(\boldsymbol{x})] \geq 1 - 3c^*\epsilon$.*

This lemma states that given an initial model that has a reasonably pure super level set, one can manage to progressively correct a large fraction of corrupted label if one run Algorithm 1 for a sufficient long time with carefully chosen parameters. The limit of Algorithm 1 will depend on the approximation ability of the neural network, which is characterized by parameter ε in Definition 2. To prove Theorem 1 using Lemma 3, it suffices to get a model which has a reliable region. This is provably achievable by training with a family of good scoring functions on PMD noisy labeled data.

4 EXPERIMENTS

We evaluate our method on both synthetic experiments and real world datasets. We first conduct synthetic experiments on two public data sets CIFAR-10 and CIFAR-100 (Krizhevsky et al., 2009). To synthesize the label noise, we first approximate the true posterior probability η using the confidence prediction of a clean neural network (trained with the original labels). We call these original labels raw labels. Then we sample $y_{\boldsymbol{x}} \sim \eta(\boldsymbol{x})$ for each instance $\boldsymbol{x}$. Instead of using raw labels, we use these sampled labels $\boldsymbol{y}$ as the clean labels, whose posterior probability is exactly $\eta(\boldsymbol{x})$ and the neural

network is the Bayes optimal classifier $\eta^* : \mathcal{X} \to \{1, \cdots, C\}$, where C is the number of classes. Note that in multi-class setting, $\eta(\boldsymbol{x})$ has a vector output and $\eta_i(\boldsymbol{x})$ is the i-th element of this vector.

Noise generation. We consider a generic family of noise. We consider not only feature dependent noise, but also hybrid noise that consists of both feature dependent noise and i.i.d. noise.

For feature dependent noise, we use three types of noise function within the PMD noise family. To make the noise challenging enough, for input $\boldsymbol{x}$ we always corrupt label from the most confident category $u_{\boldsymbol{x}}$ to the second confident category $s_{\boldsymbol{x}}$, according to $\eta(\boldsymbol{x})$. Because $s_{\boldsymbol{x}}$ is the class that confuses $\eta^*(\boldsymbol{x})$ the most, this noise will hurt the network's performance the most. Note that $y_{\boldsymbol{x}}$ is sampled from $\eta(\boldsymbol{x})$, which has quite an extreme confidence. Thus we generally assume $y_{\boldsymbol{x}}$ is $u_{\boldsymbol{x}}$. For each data $\boldsymbol{x}$, we only flip it to $s_{\boldsymbol{x}}$ or keep it as $u_{\boldsymbol{x}}$. The three noise functions are as follows:

$$\text{Type-I}: \tau_{u_{\boldsymbol{x}},s_{\boldsymbol{x}}} = -\frac{1}{2}\left[\eta_{u_{\boldsymbol{x}}}(\boldsymbol{x}) - \eta_{s_{\boldsymbol{x}}}(\boldsymbol{x})\right] + \frac{1}{2}, \quad \text{Type-II}: \tau_{u_{\boldsymbol{x}},s_{\boldsymbol{x}}} = 1 - \left[\eta_{u_{\boldsymbol{x}}}(\boldsymbol{x}) - \eta_{s_{\boldsymbol{x}}}(\boldsymbol{x})\right]^2$$

$$\text{Type-III}: \tau_{u_{\boldsymbol{x}},s_{\boldsymbol{x}}} = 1 - \frac{1}{3}\left[\left[\eta_{u_{\boldsymbol{x}}}(\boldsymbol{x}) - \eta_{s_{\boldsymbol{x}}}(\boldsymbol{x})\right]^3 - \left[\eta_{u_{\boldsymbol{x}}}(\boldsymbol{x}) - \eta_{s_{\boldsymbol{x}}}(\boldsymbol{x})\right]^2 - \left[\eta_{u_{\boldsymbol{x}}}(\boldsymbol{x}) - \eta_{s_{\boldsymbol{x}}}(\boldsymbol{x})\right]\right]$$

Notice that the noise level is determined by the $\eta(\boldsymbol{x})$ naturally and we cannot control it directly. To change the noise level, we multiply $\tau_{u_{\boldsymbol{x}},s_{\boldsymbol{x}}}$ by certain constant factor such that the final proportion of noise matches our requirement. For PMD noise only, we test noise levels 35% and 70%, meaning 35% and 70% of the data are corrupted due to the noise, respectively.

For i.i.d. noise we follow the convention and adopt the commonly used uniform noise and asymmetric noise (Patrini et al., 2017). We artificially corrupt the labels by constructing the noise transition matrix T, where $T_{ij} = P(\widetilde{y} = j|y = i) = \tau_{ij}$ defines the probability that a true label $y = i$ is flipped to j. Then for each sample with label i, we replace its label with the one sampled from the probability distribution given by the i-th row of matrix T. We consider two kinds of i.i.d. noise in this work. (1) Uniform noise: the true label i is corrupted uniformly to other classes, i.e., $T_{ij} = \tau/(\mathcal{C} - 1)$ for $i \neq j$, and $T_{ii} = 1 - \tau$, where τ is the constant noise level; (2) Asymmetric noise: the true label i is flipped to j or stays unchanged with probabilities $T_{ij} = \tau$ and $T_{ii} = 1 - \tau$, respectively.

Baselines. We compare our method with several recently proposed approaches. (1) GCE (Zhang & Sabuncu, 2018); (2) Co-teaching+ (Yu et al., 2019); (3) SL (Wang et al., 2019); (4) LRT (Zheng et al., 2020). All these methods are generic and handle the label noise without assuming the noise structures. Finally, we also provide the results by Standard method, which simply trains the deep network on noisy datasets in a standard manner.

During training, we use a batch size of 128 and train the network for 180 epochs to ensure the convergence of all methods. We train the network with SGD optimizer, with initial learning rate 0.01. We randomly repeat the experiments 3 times, and report the mean and standard deviation values.

Results. Table 1 lists the performance of different methods with three types of feature dependent noise at noise levels 35% and 70%. We observe that our method achieves the best performance across different noise settings. Moreover, notice that some of the baseline methods' performance is inferior to the standard approach. Possible reasons are that these methods behave too conservatively in dealing with noise. Thus they only make use a small subset of the original training set, which is not representative enough to grant the model good discriminative ability.

In Table 2 we show the results on datasets corrupted with a combination of feature-dependent noise and i.i.d. noise, which ends up to real noise levels ranging from 50% to 70% (in terms of proportion of corrupted labels). I.i.d. noise is overlayed on the feature dependent noise. Our method outperforms baselines under these more complicated noise patterns. In contrast, when the noise level is high like the case where we further apply additional 30% and 60% i.i.d noise, performances of a few baselines deteriorate and become worse than the standard approach.

We carry out the ablation study on the hyperparameter θ (determining the confidence threshold for label correction, see Alg. 1). In Table 4, we show that our method is robust against the choice of θ up to a wide range. Notice that here we are calculating the absolute difference of $\log f_{\widetilde{y}}(\boldsymbol{x})$ and $\log f_{h_{\boldsymbol{x}}}(\boldsymbol{x})$. As we mentioned in Section 2.2, this operation gives a good performance in practice.

Results on real-world noisy dataset. To test the effectiveness of the proposed method under real-world label noise, we conduct experiments on the Clothing1M dataset (Xiao et al., 2015). This dataset contains 1 million clothing images obtained from online shopping websites with 14 categories. The

labels in this dataset are quite noisy with an unknown underlying structure. This dataset provides 50k, 14k and 10k manually verified clean data for training, validation and testing, respectively. Following (Tanaka et al., 2018; Yi & Wu, 2019), in our experiment we discard the 50k clean training data and evaluate the classification accuracy on the 10k clean data. Also, following (Yi & Wu, 2019), we use a randomly sampled pseudo-balanced subset as the training set, which includes about 260k images. We set the batch size 32, learning rate 0.001, and adopt SGD optimizer and use ResNet-50 with weights pre-trained on ImageNet, as in (Tanaka et al., 2018; Yi & Wu, 2019).

We compare our method with the following baselines. (1) Standard; (2) Forward Correction (Patrini et al., 2017); (3) D2L (Ma et al., 2018); (4) JO (Tanaka et al., 2018); (5) PENCIL (Yi & Wu, 2019); (6) DY (Arazo et al., 2019); (7) GCE (Zhang & Sabuncu, 2018); (8) SL (Wang et al., 2019); (9) MLNT (Li et al., 2019); (10) LRT (Zheng et al., 2020). In Table 3 we observe that our method achieves the best performance, suggesting the applicability of our label correction strategy in real-world scenarios.

Table 1: Test accuracy (%) on CIFAR-10 and CIFAR-100 under different feature dependent noise types and levels. The average accuracy and standard deviations over 3 trials are reported.

Dataset	Noise	Standard	Co-teaching+	GCE	SL	LRT	**PCL**
CIFAR-10	Type-I (35%)	78.11 ± 0.74	79.97 ± 0.15	80.65 ± 0.39	79.76 ± 0.72	80.98 ± 0.80	$\mathbf{82.80 \pm 0.27}$
	Type-I (70%)	41.98 ± 1.96	40.69 ± 1.99	36.52 ± 1.62	36.29 ± 0.66	41.52 ± 4.53	$\mathbf{42.74 \pm 2.14}$
	Type-II (35%)	76.65 ± 0.57	77.34 ± 0.44	77.60 ± 0.88	77.92 ± 0.89	80.74 ± 0.25	$\mathbf{81.54 \pm 0.47}$
	Type-II (70%)	45.57 ± 1.12	45.44 ± 0.64	40.30 ± 1.46	41.11 ± 1.92	44.67 ± 3.89	$\mathbf{46.04 \pm 2.20}$
	Type-III (35%)	76.89 ± 0.79	78.38 ± 0.67	79.18 ± 0.61	78.81 ± 0.29	81.08 ± 0.35	$\mathbf{81.50 \pm 0.50}$
	Type-III (70%)	43.32 ± 1.00	41.90 ± 0.86	37.10 ± 0.59	38.49 ± 1.46	44.47 ± 1.23	$\mathbf{45.05 \pm 1.13}$
CIFAR-100	Type-I (35%)	57.68 ± 0.29	56.70 ± 0.71	58.37 ± 0.18	55.20 ± 0.33	56.74 ± 0.34	$\mathbf{60.01 \pm 0.43}$
	Type-I (70%)	39.32 ± 0.43	39.53 ± 0.28	40.01 ± 0.71	40.02 ± 0.85	45.29 ± 0.43	$\mathbf{45.92 \pm 0.61}$
	Type-II (35%)	57.83 ± 0.25	56.57 ± 0.52	58.11 ± 1.05	56.10 ± 0.73	57.25 ± 0.68	$\mathbf{63.68 \pm 0.29}$
	Type-II (70%)	39.30 ± 0.32	36.84 ± 0.39	37.75 ± 0.46	38.45 ± 0.45	43.71 ± 0.51	$\mathbf{45.03 \pm 0.50}$
	Type-III (35%)	56.07 ± 0.79	55.77 ± 0.98	57.51 ± 1.16	56.04 ± 0.74	56.57 ± 0.30	$\mathbf{63.68 \pm 0.29}$
	Type-III (70%)	40.01 ± 0.18	35.37 ± 2.65	40.53 ± 0.60	39.94 ± 0.84	44.41 ± 0.19	$\mathbf{44.45 \pm 0.62}$

Table 2: Test accuracy (%) on CIFAR-10 and CIFAR-100 under different hybrid noise types and levels. The average accuracy and standard deviations over 3 trials are reported.

Dataset	Noise	Standard	Co-teaching+	GCE	SL	LRT	**PCL**
CIFAR-10	Type-I + 30% Uniform	75.26 ± 0.32	78.72 ± 0.53	78.08 ± 0.66	77.79 ± 0.46	75.97 ± 0.27	$\mathbf{79.04 \pm 0.50}$
	Type-I + 60% Uniform	64.25 ± 0.78	55.49 ± 2.11	67.43 ± 1.43	67.63 ± 1.36	59.22 ± 0.74	$\mathbf{72.21 \pm 2.92}$
	Type-I + 30% Asymmetric	75.21 ± 0.64	75.43 ± 2.96	76.91 ± 0.56	77.14 ± 0.70	76.96 ± 0.45	$\mathbf{78.31 \pm 0.41}$
	Type-II + 30% Uniform	74.92 ± 0.63	75.19 ± 0.54	75.69 ± 0.21	75.08 ± 0.47	75.94 ± 0.58	$\mathbf{80.08 \pm 0.37}$
	Type-II + 60% Uniform	64.02 ± 0.66	59.89 ± 0.63	66.39 ± 0.29	66.76 ± 1.60	58.99 ± 1.43	$\mathbf{71.21 \pm 1.46}$
	Type-II + 30% Asymmetric	74.28 ± 0.39	73.37 ± 0.83	75.30 ± 0.81	75.43 ± 0.42	77.03 ± 0.62	$\mathbf{77.63 \pm 0.30}$
	Type-III + 30% Uniform	74.00 ± 0.38	77.31 ± 0.11	77.00 ± 0.12	76.22 ± 0.12	75.66 ± 0.57	$\mathbf{80.06 \pm 0.47}$
	Type-III + 60% Uniform	63.96 ± 0.69	56.78 ± 1.56	67.53 ± 0.51	67.79 ± 0.54	59.36 ± 0.93	$\mathbf{73.48 \pm 1.84}$
	Type-III + 30% Asymmetric	75.31 ± 0.34	74.62 ± 1.71	75.70 ± 0.91	76.09 ± 0.10	77.19 ± 0.74	$\mathbf{77.54 \pm 0.70}$
CIFAR-100	Type-I + 30% Uniform	48.86 ± 0.56	52.33 ± 0.64	52.90 ± 0.53	51.34 ± 0.64	45.66 ± 1.60	$\mathbf{60.09 \pm 0.15}$
	Type-I + 60% Uniform	35.97 ± 1.12	27.17 ± 1.66	38.62 ± 1.65	37.57 ± 0.43	23.37 ± 0.72	$\mathbf{51.68 \pm 0.10}$
	Type-I + 30% Asymmetric	45.85 ± 0.93	51.21 ± 0.31	52.69 ± 1.14	50.18 ± 0.97	52.04 ± 0.15	$\mathbf{56.40 \pm 0.34}$
	Type-II + 30% Uniform	49.32 ± 0.36	51.99 ± 0.75	53.61 ± 0.46	50.58 ± 0.25	43.86 ± 1.31	$\mathbf{60.01 \pm 0.63}$
	Type-II + 60% Uniform	35.16 ± 0.05	25.91 ± 0.64	39.58 ± 3.13	37.93 ± 0.22	23.05 ± 0.99	$\mathbf{49.35 \pm 1.53}$
	Type-II + 30% Asymmetric	46.50 ± 0.95	51.07 ± 1.44	51.98 ± 0.37	49.46 ± 0.23	52.11 ± 0.46	$\mathbf{61.43 \pm 0.33}$
	Type-III + 30% Uniform	48.94 ± 0.61	49.94 ± 0.44	52.07 ± 0.35	50.18 ± 0.54	42.79 ± 1.78	$\mathbf{60.14 \pm 0.97}$
	Type-III + 60% Uniform	34.67 ± 0.16	22.89 ± 0.75	36.82 ± 0.49	37.65 ± 1.42	22.81 ± 0.72	$\mathbf{50.73 \pm 2.16}$
	Type-III + 30% Asymmetric	45.70 ± 0.12	49.38 ± 0.86	50.87 ± 1.12	48.15 ± 0.90	50.31 ± 0.39	$\mathbf{54.56 \pm 1.11}$

Table 3: Test accuracy (%) on Clothing1M.

Method	Standard	Forward	D2L	JO	PENCIL	DY	GCE	SL	MLNT	LRT	**PCL**
Accuracy	68.94	69.84	69.47	72.23	73.49	71.00	69.75	71.02	73.47	71.74	**74.02**

Table 4: The effect of θ on the performance. We use CIFAR-10 with 35% feature-dependent noise.

$\exp(\theta)$	0.2	0.3	0.4	0.5
Type-I Noise	83.33	83.04	82.66	82.94
Type-II Noise	81.84	81.18	81.09	81.24
Type-III Noise	81.79	81.75	81.98	82.06

5 CONCLUSION

We propose a novel family of feature-dependent label noise that is much more general than traditional i.i.d. noise pattern. Building on such noise assumption, we propose the first data-recalibrating method with the theoretical guarantee that it converges to an ideal classifier. We show in practice that our method outperforms various baselines on various feature-dependent noise satisfying our assumption.

REFERENCES

Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Unsupervised label noise modeling and loss correction. In *ICML*, 2019.

Nontawat Charoenphakdee, Jongyeong Lee, and Masashi Sugiyama. On symmetric losses for learning from corrupted labels. In *ICML*, 2019.

Jiacheng Cheng, Tongliang Liu, Kotagiri Ramamohanarao, and Dacheng Tao. Learning with bounded instance- and label-dependent label noise. In *ICML*, 2020.

Hendrycks Dan, Lee Kimin, and Mazeika Mantas. Using pre-training can improve model robustness and uncertainty. In *ICML*, pp. 2712–2721, 2019.

Jun Du and Zhihua Cai. Modelling class noise with symmetric and asymmetric distributions. In *AAAI*, pp. 2589–2595, 2015.

Wei Gao, Bin-Bin Yang, and Zhi-Hua Zhou. On the resistance of nearest neighbor to random noisy labels. *arXiv preprint arXiv:1607.07526*, 2016.

Aritra Ghosh, Naresh Manwani, and P.S. Sastry. Making risk minimization tolerant to label noise. *Neurocomput*, 160:93–107, 2015.

Aritra Ghosh, Himanshu Kumar, and PS Sastry. Robust loss functions under label noise for deep neural networks. In *AAAI*, 2017.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor W. Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *NeurIPS*, pp. 8536–8546, 2018.

W Hu, Z Li, and D Yu. Simple and effective regularization methods for training on noisily labeled data with generalization guarantee. In *ICLR*, 2020.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *ICML*, pp. 2309–2318, 2018.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. URL `https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf`.

Junnan Li, Yongkang Wong, Qi Zhao, and Mohan S Kankanhalli. Learning to learn from noisy labeled data. In *CVPR*, pp. 5051–5059, 2019.

Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li-Jia Li. Learning from noisy labels with distillation. In *ICCV*, pp. 1928–1936, 2017.

Xingjun Ma, Yisen Wang, Michael E. Houle, Shuo Zhou, Sarah M. Erfani, Shu-Tao Xia, Sudanthi N. R. Wijewickrema, and James Bailey. Dimensionality-driven learning with noisy labels. In *ICML*, pp. 3361–3370, 2018.

Naresh Manwani and PS Sastry. Noise tolerance under risk minimization. *IEEE transactions on cybernetics*, 43(3):1146–1151, 2013.

Aditya Krishna Menon, Brendan Rooyen, and Nagarajan Natarajan. Learning from binary labels with instance-dependent noise. *Mach. Learn.*, 107(8–10):1561–1595, 2018.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *NeurIPS*, pp. 1196–1204, 2013.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *CVPR*, pp. 2233–2241, 2017.

Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. In *ICLR Workshop*, 2014.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *ICML*, pp. 4331–4340, 2018.

Yanyao Shen and Sujay Sanghavi. Learning with bad training data via iterative trimmed loss minimization. In *International Conference on Machine Learning*, pp. 5739–5748. PMLR, 2019.

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. In *NeurIPS*, pp. 1917–1928, 2019.

Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. Joint optimization framework for learning with noisy labels. In *CVPR*, pp. 5552–5560, 2018.

Sunil Thulasidasan, Tanmoy Bhattacharya, Jeff Bilmes, Gopinath Chennupati, and Jamal Mohd-Yusof. Combating label noise in deep learning using abstention. In *ICML*, 2019.

Arash Vahdat. Toward robustness against label noise in training deep discriminative neural networks. In *NeurIPS*, pp. 5596–5605, 2017.

Brendan Van Rooyen, Aditya Menon, and Robert C Williamson. Learning with symmetric label noise: The importance of being unhinged. In *NeurIPS*, pp. 10–18, 2015.

Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge J. Belongie. Learning from noisy large-scale datasets with minimal supervision. In *CVPR*, pp. 6575–6583, 2017.

Yisen Wang, Weiyang Liu, Xingjun Ma, James Bailey, Hongyuan Zha, Le Song, and Shu-Tao Xia. Iterative learning with open-set noisy labels. In *CVPR*, pp. 8688–8696, 2018.

Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *CVPR*, pp. 322–330, 2019.

Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *CVPR*, pp. 2691–2699, 2015.

Yilun Xu, Peng Cao, Yuqing Kong, and Yizhou Wang. L_dmi: A novel information-theoretic loss function for training deep nets robust to label noise. In *Advances in Neural Information Processing Systems*, pp. 6225–6236, 2019.

Yan Yan, Rómer Rosales, Glenn Fung, Ramanathan Subramanian, and Jennifer Dy. Learning from multiple annotators with varying expertise. *Machine learning*, 95(3):291–327, 2014.

Kun Yi and Jianxin Wu. Probabilistic end-to-end noise correction for learning with noisy labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7017–7025, 2019.

Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *ICML*, pp. 7164–7173, 2019.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *ICLR*, 2017.

Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *NeurIPS*, pp. 8778–8788, 2018.

Songzhu Zheng, Pengxiang Wu, Aman Goswami, Mayank Goswami, Dimitris Metaxas, and Chao Chen. Error-bounded filtering of label noise with deep neural networks. In *ICML*, 2020.

A APPENDIX

Lemma 1 (One round purity improvement)**.** *Assume 1), and an f such that there exists a level set $L(e,\eta)$ pure to f with $3\epsilon \leq e < \frac{1}{2}$. Let $\tilde{\eta}_{new}(\boldsymbol{x}) = y_{f(\boldsymbol{x})}$ if $|f(\boldsymbol{x}) - 1/2| \geq e$ and $\tilde{\eta}(\boldsymbol{x})$ if $|f(\boldsymbol{x}) - 1/2| < e$, and assume $f_{new} = \arg\min_{h\in\mathcal{H}} E_{(x,\tilde{y})\sim D(\boldsymbol{x},\tilde{\eta}_{new}(\boldsymbol{x}))} Loss(h(\boldsymbol{x}),\tilde{y})$. Let $e_{new} = \min\{e | e > 0,\ L(e,\eta)$ is pure for $f_{new}\}$. Then $\frac{1}{2} - e_{new} \geq (1 + \frac{\varepsilon}{\alpha\ell})(\frac{1}{2} - e)$.*

Proof: We analyze the case where $\eta(\boldsymbol{x}) > \frac{1}{2}$, the analysis on the other side can be derived similarly. Due to the fact that there exists level set $L(e,\eta)$ pure to f, we have $\forall x$ with $e \leq |\eta(x) - \frac{1}{2}|$,

$$\mathbb{E}_{(z,\tilde{y})\sim(D,\tilde{\eta}_{new}(z))}[\mathbb{1}(\tilde{y}_z \neq \eta^*(z)) | |\eta(z) - \frac{1}{2}| \geq |\eta(x) - \frac{1}{2}|] = 0$$

Now consider x where $e - \gamma \leq |\eta(\boldsymbol{x}) - \frac{1}{2}|$. Due to the (c_*, c^*)-bounded distribution property of D we have

$$\begin{aligned}
&\mathbb{E}_{(z,\tilde{y})\sim(D,\tilde{\eta}_{new}(z))}[\mathbb{1}(\tilde{y}_z \neq \eta^*(z)) | |\eta(z) - \frac{1}{2}| \geq |\eta(x) - \frac{1}{2}|] \\
=&\mathbb{P}_z[\tilde{y}_z \neq \eta^*(z) | |\eta(z) - \frac{1}{2}| \geq |\eta(x) - \frac{1}{2}|] \\
=&\frac{\mathbb{P}_z[\tilde{y}_z \neq \eta^*(z), |\eta(z) - \frac{1}{2}| \geq |\eta(x) - \frac{1}{2}|]}{\mathbb{P}_z[|\eta(z) - \frac{1}{2}| \geq |\eta(x) - \frac{1}{2}|]} \\
\leq&\underbrace{\frac{\mathbb{P}_z[\tilde{y}_z \neq \eta^*(z), \frac{1}{2} + e - \gamma \leq \eta(z) \leq \frac{1}{2} + e]}{\mathbb{P}_z[|\eta(z) - \frac{1}{2}| \geq |\eta(x) - \frac{1}{2}|]}}_{\leq \frac{c^*\gamma}{c_*(\frac{1}{2} - e + \gamma)}} \\
&+\underbrace{\frac{\mathbb{P}_z[\tilde{y}_z \neq \eta^*(z), \frac{1}{2} + e \leq \eta(z)]}{\mathbb{P}_z[|\eta(z) - \frac{1}{2}| \geq |\eta(x) - \frac{1}{2}|]}}_{=0} \\
=&\frac{c^*\gamma}{c_*(\frac{1}{2} - e + \gamma)}
\end{aligned} \tag{3}$$

If $\frac{\varepsilon}{\ell\alpha}(\frac{1}{2} - e) \leq \gamma \leq \frac{2\varepsilon}{\ell\alpha}(\frac{1}{2} - e)$, the impurity in super level set $\frac{1}{2} + e - \gamma$ is at most $\frac{2\epsilon}{\alpha}$. The (α,ε)-level set consistency condition implies $|f_{new}(\boldsymbol{x}) - \eta(\boldsymbol{x})| \leq 2\epsilon$ for $\boldsymbol{x}$ s.t. $e - \gamma \leq |\eta(\boldsymbol{x}) - \frac{1}{2}|$. If $e \geq 3\epsilon$, $f_{new}(\boldsymbol{x})$ will give same label as $\eta(\boldsymbol{x})$ thus $(e - \gamma, \eta)$-level set becomes pure for f. Meanwhile the choice of γ ensures that $\frac{1}{2} - e_{new} \geq (1 + \frac{\varepsilon}{\ell\alpha})(\frac{1}{2} - e)$. □

Lemma 2. *Suppose for a given function f_0 there exists a level set $L(e_0,\eta)$ which is pure for f_0. Given $T_0 < 1/2$, after running Algorithm 1 for $m \geq \frac{\ell\alpha}{\varepsilon}\log(\frac{2T_0}{1-2e_0})$ rounds, there exists a level set $L(\frac{1}{2} - T_0, \eta)$ that is pure for f_0.*

Proof:
The proof follows from the fact that each round of flipping improves purity in a factor of $(1 + \frac{\varepsilon}{\ell\alpha})$. To obtain at least a T_0 pure region, it suffices to repeat the flipping step for $m \geq \frac{\ell\alpha}{\varepsilon}\log(\frac{T_0}{1-2e_0})$ rounds. □

Lemma 3. *Assuming 1) given f_0, there exists a level set $L(e_0,\eta)$ pure to f_0 2) D is (c_*, c^*)-bounded distribution 3) $\mathcal{H}$ is (α,ε)-level set consistent. If one runs Algorithm 1 with $T_0 < \frac{1}{2} - e_0$, $m \geq \frac{\ell\alpha}{\varepsilon}\log(\frac{2T_0}{1-2e_0})$, $N \geq m + \frac{1}{\beta}\log(\frac{1-6\varepsilon}{2T_0})$, $T_{end} \leq \frac{1}{2} - 3\epsilon$ and $\frac{\varepsilon}{\alpha\ell} \leq \beta \leq \frac{2\varepsilon}{\alpha\ell}$, we have $\mathbb{P}_{x\sim D}[y_{f_{final}(\boldsymbol{x})} = \eta^*(\boldsymbol{x})] \geq 1 - 3c^*\epsilon$.*

Proof:
The proof can be done by combining Lemma 1 and Lemma 2. In the first m iterations, by Lemma 2, we can guarantee a level set $(\frac{1}{2} - T_0, \eta)$ pure to f. In rest of the iteration we ensure the level set $|\eta(x) - \frac{1}{2}| \geq \frac{1}{2} - T$ is pure. We increase T by a reasonable factor of β to avoid incurring to many corrupted label meanwhile ensuring enough progress in purifying i.e., $\frac{\varepsilon}{\alpha\ell} \leq \beta \leq \frac{2\varepsilon}{\alpha\ell}$ so that in the level set $|\eta(x) - \frac{1}{2}| \geq \frac{1}{2} - T$, we have $|f(x) - \eta(x)| \leq 3\varepsilon$. This condition ensure the correctness of

flipping when $T \leq \frac{1}{2} - 3\varepsilon$. The purity can not be improved once $T \geq \frac{1}{2} - 3\varepsilon = T_{end}$ since there is no guarantee that $f(x)$ is has consistent label with $\eta(x)$ when $|\eta(x) - \frac{1}{2}| < 3\varepsilon$ and $|\eta(x) - f(x)| \leq 3\varepsilon$. By (c_*, c^*)-bounded assumption on D, its mass of impure 3ε level set region is at most $3c^*\epsilon$. □

Theorem 1. *Under assumption 1, for any noise τ which is PMD with margin t_0, define $e_0 = max(t_0, \frac{\alpha+\varepsilon}{1+2\alpha})$. Then the output of Algorithm 1 with f as above and with the following initializations: a) $T_0 < \frac{1}{2} - e_0$, b) $m \geq \frac{\ell\alpha}{\varepsilon}\log(\frac{2T_0}{1-2e_0})$, c)$N \geq m + \frac{1}{\beta}\log(\frac{T_0}{3\varepsilon})$, d) $T_{end} \leq 3\epsilon$ and e) $\beta \leq \frac{\varepsilon}{\alpha\ell}$, we have*

$$\mathbb{P}_{x\sim D}[y_{f_{final}(\boldsymbol{x})} = \eta^*(\boldsymbol{x})] \geq 1 - 3c^*\epsilon.$$

Proof: The proof is based on Lemma 3 plus a verification of existence f_0 to that there exists a Pure (e_0, f, η)-level set. Let:

$$\tau(\boldsymbol{x}) = \begin{cases} \tau_{10}(\boldsymbol{x}) & \eta(\boldsymbol{x}) \geq 1/2 \\ \tau_{01}(\boldsymbol{x}) & \eta(\boldsymbol{x}) < 1/2 \end{cases}$$

In the level set $|\eta(x) - \frac{1}{2}| \geq e_0, \mathbb{P}_z[\tilde{y}_z \neq \eta^*(z)| \, |\eta(z) - \frac{1}{2}| \geq e_0] \leq \frac{1}{2} - e_0 + \tau(z)$. By (α, ε)-level set consistency, it suffices to satisfy $\alpha(\frac{1}{2} - e_o + \tau) + \varepsilon \leq e_0$ to ensure that $f(x)$ has the same prediction with $\eta(x)$ when $|\eta(x) - \frac{1}{2}| \geq e_0$. By polynomial level set diminishing noise, $\tau(x) \leq \frac{1}{2} - e_0$ if $e_0 > t_0$ thus by choosing $e_0 = max(t_0, \frac{\alpha+\varepsilon}{1+2\alpha})$ one can ensure that initial f_0 has a Pure (e_0, f_0, η)-level set. The rest of the proof follows from Theorem 1. □